

**Automation Case Study:
Election Night Reporting**

Daniel Hendrix

Information Technology Services Specialist III,
SC State Election Commission

February 5th, 2018

Contents

The Case for Automation	3
Election Night Reporting	4
Components of ENR	4
Gap Analysis	5
Current State	5
Desired State	7
Proposed Solution	8
Data Collection Process	9
Development of Rulesets	10
Office Titles	10
Aggregation Numbers	11
Map Files	12
Primaries	12
Additional Improvements	13
Data Validation	14
Further Automation and Future Goals	14
Evaluation of Final Solution	15
Future Application and Recommendations	15

The Case for Automation

As is the case with most agencies throughout the state, there are functions within the South Carolina State Election Commission, or SEC, that are very time-consuming and repetitive. These functions, while often quite simple in terms of their execution, also require a high degree of accuracy. However, as is the case with any repetitive task, accuracy can become difficult to maintain, and, in the worst cases, this may even negate the entire process. These processes, if unchecked, can easily overwhelm any business entity or agency, so it is important to identify them and actively work to improve them. One possible improvement is process automation.

As small agencies, such as the State Election Commission, require that all employees serve multiple functions, identifying opportunities to automate specific job duties is imperative. A properly implemented system accomplishes several things; it increases the speed at which a job can be performed, while also increasing accuracy. In many circumstances, it can even improve employee morale by freeing a worker from duties that are repetitive and mindless and enabling them to perform job functions that require critical thinking and intuition; that is, duties that are better suited for a human.

However, in order for a process to be effectively automated, the process designer must carefully consider all potential variations in the process and data. Issues must be anticipated and steps taken to mitigate them before they occur. Existing data, if available, must be carefully analyzed, as it will help to create a blueprint for the process, as well as serving as a means of verifying the accuracy of the system. Ultimately, failure to automate a process properly may result in the need for human intervention, or worse, output that is incorrect, which, if not detected, could be detrimental to the process or agency.

Election Night Reporting

One function within the State Election Commission that could potentially benefit from automation is that of Election Night Reporting. Election Night Reporting, or, ENR, is an online system used for tabulating, monitoring, and displaying election results on the evening of a statewide or other major election in South Carolina.

Typically, in a municipal, school board, or any other election that is isolated to a small number of counties, election results are posted or displayed at the respective county elections offices. However, in a large election, such as a gubernatorial or presidential election, in addition to being posted locally, each county's results data is uploaded to the ENR website, which is used to tally results live, and allows them to be displayed to a much larger audience.

Because these results are live, rules must be applied to the data to ensure that all information, including votes, candidate names, and office titles display correctly as they are uploaded to the system, as the data may be viewed and shared by people and news sources all over the world. If these rulesets are incorrect, they can potentially cast doubt on the election process in the state, so it is imperative that they are accurate and complete prior to election night.

Components of ENR

Election results files with the votes zeroed out, also known as “zero” files, are the foundation of the ENR process. These files are collected from each of the five database, or ballot, builders at the SEC, as well as the eight counties who build their own. These files contain several important data fields pertaining to the ENR process, including vote counts, precinct names, office title, and candidate name.

In addition, the GIS division of the Department of Administration, formerly known as the Office of Research and Statistics, produces map files which are used for some of the data visualization aspects of ENR. Each county's folder contains four map files, however, only two of these are used for Election Night Reporting. One of these files contains the shape data, or outline, for each of the precincts within a county. The other contains the precinct names as they pertain to these shapes. In order for votes to tally correctly, the office title and candidate names must be consistent across all participating counties. In addition, in order for map data to display correctly, the precinct names contained within the zero file must match those in the map file. Rules must be written to account for inconsistencies within these datasets so that all data on election night displays correctly.

Gap Analysis

Current State

The Election Night Reporting process, as of May 2014, consisted of a spreadsheet which performed some rudimentary rule generation. This spreadsheet contained a series of empty cells in which data could be typed or pasted, interspersed with the XML code needed to encapsulate each of these strings. The output of the spreadsheet was a concatenation of each of the XML formatted rulesets that could then be copied and pasted into a template and uploaded to the ENR site.

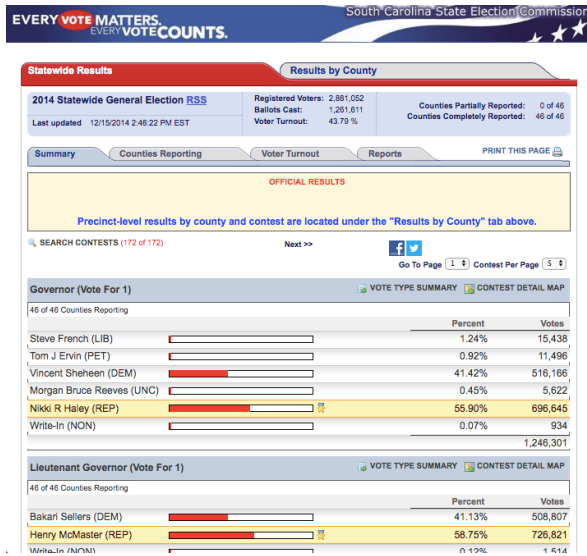


Figure 1. In November of 2014, in addition to the Governor, all of South Carolina’s one-hundred-and-twenty-four State House offices, as well as all seven U.S. House offices were up for re-election. This resulted in 172 contests on the state page alone.

Use of this method poses many potential issues. First, this method is extremely time consuming. Each value must be manually copied from the zero file and/or map file, and pasted into the correct cell on the spreadsheet. If a value is pasted into the wrong cell, it will result in election results not displaying correctly. In addition, each ruleset is only truly valid for one county, meaning that this process must be repeated up to forty-six times with the utmost accuracy, while still maintaining an awareness of the anomalies contained within each county.

Finally, both the map and zero files may contain unusual or non-readable system characters, such as a single quotation mark instead of an apostrophe. While these two characters may appear the same, if these differences are not accounted for, the data will not display correctly, and, sometimes, even copying and pasting directly will not resolve the issue.

Most elections databases are built internally at the SC State Election Office. However, due to the potential burden that some larger counties would place on the SEC if their elections were built internally, there are eight counties that have been authorized to build their own databases. Though there are standards in place, due to the complexities of the databases and subtle variations in naming conventions, not all data that is exported on election night from each of the 46 counties is formatted consistently. For example, in the case of districts, databases may refer to them as “DISTRICT”, “Dist”, “Dist.”, or even “District7”, among others.

Similarly, precinct names within the zero file must also be identical to those contained within the map files provided by the state's GIS division. As is the case with office and candidate names, even small variations will prevent data from displaying correctly, which can result in areas of the county maps being marked as not participating or displaying an incorrect name. In addition to minor variations in punctuation, there are also cases in which a precinct may be abbreviated, such as the use of "X roads" to mean "Crossroads".

Another important consideration of this process is that aggregation numbers must be assigned to all offices in order for them to calculate and display correctly. These aggregation numbers must be entered carefully and consistently across all rulesets, as being even one digit off will move votes for one office into votes for another. Despite the spreadsheet assisting in terms of the formatting, this is still a very manual process that requires a lot of time and human intervention.

Even if rules are created for all possible combinations of the data, it is extremely likely that there will still be errors in the data on election night. While some of these errors are less significant than others, and despite the fact that all of these errors will eventually be identified and resolved, it is important to keep the number of errors to a minimum. A majority of these errors can be corrected very quickly and easily during or between results uploads, but if more than a handful of counties require rules to be changed or added on election night, it can quickly cause issues with the reporting process and the accuracy of the results.

Desired State

Ultimately, the goal of the State Election Commission in terms of Election Night Reporting is to provide the public with the most accurate election results possible, and to report them in a timely manner. While diligence under the existing system adequately accomplishes those aspects of ENR,

improvements may certainly be made to make the overall process more efficient and less prone to errors.

Key Factors in Desired State as it Pertains to ENR:

- Maintain or increase accuracy of results
- Increase efficiency of process
- Reduce or eliminate errors to be corrected on or after election night
- Account for all possible variations in datasets between counties and organizations
- System should be easy to adjust to account for future legislation or changes
- Increase public perception and trust in the elections process
- Provide a clear correlation between data in election management system and ENR
- Account for all circumstances pertaining to party and office titles
- Create standards for output data to ENR
- Solution should be usable for all types of elections, including general elections, special elections, and primaries
- Solution should be adaptable to other agency functions where available

Proposed Solution

Due to the data-driven nature of the process, Election Night Reporting would likely be a prime candidate for automation. Development of this process would require a thorough knowledge of the interplay between numerous aspects of the voting system, election management system, and the elected offices in the state. It would also require that any resulting data or process be able to interface with the current Election Night Reporting system.

Data Collection Process

The first step in creating a system that could account for all elected offices in the state was to identify what those offices were. Because of his expertise on the functioning of state offices throughout the state, I spoke with Chris Whitmire, the Director of Public Information and Training at the SC State Election Commission. While he would later prove valuable in some of the rule generation aspects of the process, he directed me to our voter registration and election management system, or VREMS, for the full list.

The offices which were identified as the primary focus of the automation process were federal, state-level, and multi-county offices. (Fig. 2) While there are significantly more county-level offices than state or federal offices, single-county offices only display on their specific county page within ENR. Because of this, these contests do not need to be standardized nor have their results aggregated. Once the list of offices was established, the next step was to examine the existing results files to look for variations in the naming of these offices. As they contained the greatest number of these offices, and thus, likely the greatest number of variations, the 2014 general election files were used as the starting point for the office name standardization rules. Once again, Mr. Whitmire

Office Name on Ballot	Ballot Sort Order
President and Vice President	1
President	1
Governor	5
Lieutenant Governor	20
Secretary of State	25
State Treasurer	30
Attorney General	35
Comptroller General	40
State Superintendent of Education	45
Adjutant General	50
Commissioner of Agriculture	55
U.S. Senate	60
U.S. Senate 2	61
U.S. House of Representatives	65
State Senate	70
State House of Representatives	75
Solicitor Circuit 1	80

Figure 2. Listing of all Federal and State-level office titles from the voter registration system, or VREMS.

was consulted as to the correct or desired naming and order of these offices for ENR. As, with each iteration of the rules, I encountered circumstances with which I was unfamiliar, I continued to look to Mr. Whitmire for guidance throughout the remainder of the rule generation process.

Development of Rulesets

Office Titles

As the original ENR process was built upon it, and because it allowed for relatively simple manipulation of large amounts of data, Microsoft Excel would provide the foundation for the rule generation process.

A new spreadsheet was created in which office titles from VREMS and the results files were copied and pasted. Simplified versions of these titles were placed into two columns, and formulas were written to convert one to the other, taking into account the variations between the different offices, with a third column containing the original value for comparison. However, in addition to differences in abbreviation and spelling, because “Senate” and “House of Representatives” are present in both state and federal government, formulas had to be made to account for these differences as well.

Once some basic rules were in place, the next issue would be accounting for the district number of some offices. Rules were added, and the existing rule altered, to identify, remove, and then re-append the district number to these offices.

Now that a framework had been created to account for the small sample of offices that had been collected from the results files, I created a new sheet and pasted in a county’s entire results file. Results files utilize fixed-width columns, meaning that every field, including things like office title and candidate name are limited to a specific number of characters. The results file was then parsed

based on those field lengths, resulting in, among other things, a list of offices that could be present in an election. After a series of tweaks and changes to these rules, I added in the XML code from the original spreadsheet, and was now able to create rules based on office title much more quickly and easily, though the process still required copying, pasting, dividing up the results file, and removing duplicates, and then copying and pasting the resulting rule into the same template as before.

Aggregation Numbers

In addition to office titles, an equally important element of the ENR process is that of aggregation rules. These rules establish the order in which contests are displayed on ENR, and also ensure that all votes for a particular office or candidate are calculated together. Aggregation rules apply to all offices that will appear on the state page of ENR, so each county participating in a contest must have the same aggregation number for that office.

To establish aggregation rules for each office, I referenced the hierarchy from the VREMS data and assigned each office a value. However, because many offices also include a district number, that had to be accounted for so that all offices would appear sequentially. This led me to adjust my process. I once again consulted both Mr. Whitmire and VREMS to learn how many districts there were for each of these offices. This also led me to discovering the fact that South Carolina has two U.S. Senate seats which are staggered, so that one is up for election every three years. These would all be things that I would have to account for in my formulas.

Based on my research, I was now able to factor in things such as district number into my aggregation decisions. I decided on a numbering scheme, and, once again, referred to my results files for comparison. Using the list of office titles that I had extracted previously, I applied the new

aggregation rules and compared the results to the original data. After a few minor changes, I was able to produce aggregation rules using a process very similar to that which I had developed with the office titles, but one which was able to factor in those office titles as well.

Map Files

The final component of ENR to be addressed was that of map files. Similar to the process used with office titles, the precinct names column was extracted from the results data and pasted in for comparison to the names within the map file. Due to the irregular nature of precinct names, both sets of names were sorted alphabetically, with manual checks and manipulation to account for the differences. This resulted in lists that were largely correct, and the remaining errors were relatively easy to rectify since precinct names only affect one county.

Primaries

As one of my stated goals was a solution that would work for all types of elections, that would also include primaries, a factor of which I had not accounted for previously. Because primaries are conducted by party, there was a possibility that two parties could hold primaries for the same office. Because aggregation numbers must be unique, and because the order in which parties are displayed on ENR rotates, rules would need to be created and changed to account for this. I had to update my aggregation numbering scheme again, this time building in additional buffers as well to account for any other unforeseen changes.

Additional Improvements

With most of the factors that would be involved across the spectrum of elections now identified, I wanted to focus on automating the process further. Now, equipped with a better understanding of the data files, the offices, and the process in general, I decided to focus on writing formulas that would parse the data automatically, instead of requiring the user to select the data and remove duplicates prior to processing it.

In addition to the rule generation sheets, I added two blank sheets into which the results and map files could be pasted. Leveraging the nature of the fixed-width files, I was able to write a formula that would extract all office or precinct names from their respective columns and place those values into specific cells for later manipulation. Next, I utilized formulas that would examine those columns and remove duplicates, resulting in columns containing both the unique office titles and precinct names within a county.

Once the unique office titles were extracted, the original code was manipulated to factor in this new layout, and additional rules were made to clean up and standardize the data. New aggregation rules were developed, and the office title and aggregation portions were now highly automated. Due to the methods used, these formulas were not terribly efficient, at least in terms of computers, but they were much more efficient than the previous method.

Finally, the precinct names portion was upgraded to the same standard. As with office titles, the entire file could now be pasted into a sheet and the data extracted. Precinct names had any numbers and punctuation removed. Formulas were written to compare values from the map file and the results file, and any item without a match could be easily identified.

Override columns were added to both the office and map data, so that any item that was misidentified could be corrected prior to exporting the data without requiring a change in the code. Rules for formatting were added, as were rules to ignore all offices aside from those listed above.

Data Validation

With each iteration and modification of the software, output data was compared to the original files and checked for integrity. This helped to ensure that the rules were functioning properly, and that the data would display as intended.

Further Automation and Future Goals

Despite all of these improvements, the process still required far too much human intervention to be deemed “automated.” Though the rules were largely generated automatically, all ninety-two files, consisting of forty-six results files, and forty-six map files, still needed to be copied and pasted in and out manually, and there was still room for human error.

From here, the focus shifted to finding a method that would open and parse a results file automatically, ideally with the goal of having the program scan folders, import the appropriate files, and produce rulesets with little to no intervention.

Through research of different programming languages, it became apparent that Microsoft’s Visual Basic for Applications, or VBA, would be the best fit for this application. In addition, due to the prevalence of Microsoft Office within our agency and the business realm at large, this choice would also be the easiest to adapt to the changing needs of the agency or the framework.

Evaluation of Final Solution

While preliminary versions of this tool have already shown significant decreases in the time required to produce rulesets for ENR, the less human intervention there is in the process, the fewer opportunities there will be to evaluate the accuracy of the solution. Once the rule generation process is fully automated, the accuracy of the algorithms and formulas will become critical. As a result, it is advisable to spot check the automatically generated rulesets against unadulterated results files for consistency in office titles and to ensure that the tool is interpreting the data correctly.

In addition, test data should also continue to be uploaded to ENR prior to an election. This remains an important part of ensuring that results appear correctly on election night. Upon uploading test data, all state-level offices should be inspected to ensure that the correct number of counties are displayed for each office. In addition, all county maps should be viewed to verify that all participating precincts have processed correctly.

Future Application and Recommendations

At present, I have developed basic code to import a results file using a system dialogue box, as well as the inclusion of what appears to be a more efficient method of parsing files and removing duplicate data. If this proves successful, current code will need to be adapted to the change.

As the files are approaching the possibility of being opened and processed automatically, the process should also possess the ability to produce rulesets and close automatically. Ideally, this process will be able to be initiated using the first file in a series, and proceed sequentially until all remaining files have been processed.

As mentioned earlier in the context of precinct names, the software should be able to examine data in which a match is not readily available, and compare it to the other remaining options, resulting in at least a reasonable match, so that user intervention is minimal, if it is required at all.

As it stands, this software may be incorporated into the existing Election Night Reporting process. Procedures may be easily developed to outline the use of the software and instructions on making adjustments.

Where applicable, this, or similar, framework should be adapted to any and all other processes in which automation may benefit the employee, the agency, and ultimately, the citizens of the State of South Carolina.